

NASA/CR-1998-206926  
ICASE Report No. 98-14



## **A Preconditioning Method for Shape Optimization Governed by the Euler Equations**

*Eyal Arian*  
*ICASE*

*Veer N. Vatsa*  
*NASA Langley Research Center*

*Institute for Computer Applications in Science and Engineering*  
*NASA Langley Research Center*  
*Hampton, VA*

*Operated by Universities Space Research Association*



National Aeronautics and  
Space Administration

Langley Research Center  
Hampton, Virginia 23681-2199

Prepared for Langley Research Center  
under Contract NAS1-19480

---

February 1998

# A PRECONDITIONING METHOD FOR SHAPE OPTIMIZATION GOVERNED BY THE EULER EQUATIONS \*

EYAL ARIAN <sup>†</sup> AND VEER N. VATSA <sup>‡</sup>

**Abstract.** We consider a classical aerodynamic shape optimization problem subject to the compressible Euler flow equations. The gradient of the cost functional with respect to the shape variables is derived with the adjoint method at the continuous level. The Hessian (second order derivative of the cost functional with respect to the shape variables) is approximated also at the continuous level, as first introduced by Arian and Ta'asan (1996). The approximation of the Hessian is used to approximate the Newton step which is essential to accelerate the numerical solution of the optimization problem. The design space is discretized in the maximum dimension, i.e., the location of each point on the intersection of the computational mesh with the airfoil is taken to be an independent design variable. We give numerical examples for 86 design variables in two different flow speeds and achieve an order of magnitude reduction in the cost functional at a computational effort of a full solution of the analysis partial differential equation (PDE).

**Key words.** airfoil, Euler, Hessian, optimal shape, preconditioning.

**Subject classification.** Applied and Numerical Mathematics

**1. Introduction.** In the last decade there is a growing interest in optimal shape design in conjunction with fluid dynamics. The new interest in this classical field is due to advances in computer performance and improvements in algorithms for the numerical solution of the flow equations, which makes it practical to employ computational fluid dynamics (CFD) codes for shape optimization.

Historically, the inverse problem of designing a two-dimensional aerodynamical shape profile to attain a desired pressure distribution was first studied by Lighthill [2]. However, in general, the desired solution of the flow equations on the wing profile is not known in advance and a more difficult optimization problem has to be defined. The resulting optimization problems are extremely computationally expensive since it is necessary to re-solve the flow equations in each optimization step. Derivative based optimal shape methods for problems governed by flow were studied by Pironneau [3] where the adjoint method was used for the minimum-drag problem in Stokes flow and subsequently in flows governed by the incompressible Navier-Stokes equations. Jameson [4] applied the adjoint method for optimal shape design of an airfoil using the Euler equations. Since then there were numerous publications on shape optimization using CFD, most of which are contained, or referred to, in Mandel et al. [5] and in the proceedings of the AIAA and ECCOMAS meetings. Recently, Soemarwoto [6] computed optimal shapes of airfoils using the compressible Navier-Stokes equations, and Jameson et al. [7] obtained results for a three dimensional wing.

A major effort in that field is in the reduction of the enormous computational effort required to solve aerodynamic optimization problems. There are mainly two reasons for the high computational cost; one reason is the high computational cost required by each cost functional evaluation, and the second reason

---

\*This research was supported by the National Aeronautics and Space Administration under NASA Contract No. NAS1-19480 while the first author was in residence at the Institute for Computer Applications in Science and Engineering (ICASE), M/S 403, NASA Langley Research Center, Hampton, VA 23681-0001.

<sup>†</sup> ICASE, M/S 403, NASA Langley Research Center, Hampton, VA 23681-0001 (email: [arian@icase.edu](mailto:arian@icase.edu)).

<sup>‡</sup> Aerodynamic and Acoustic Methods Branch, M/S 128, NASA Langley Research Center, Hampton, VA 23681-0001 (email: [v.n.vatsa@larc.nasa.gov](mailto:v.n.vatsa@larc.nasa.gov)).

is the combination of an highly ill-conditioned cost functional Hessian and large design space (hundreds of design variables).

We give here a brief summary of the effort done so far to accelerate the numerical solution of such problems. Jameson [4] applied multigrid methods to accelerate the solution of the analysis and adjoint equations, while performing the optimization steps only on the finest grid; this shortens the cost functional evaluation time. Jameson used a smoothing procedure on the gradients to reduce the ill-conditioning difficulty. Ta’asan [8] has proposed the “one-shot” multigrid method where the design space is finite-dimensional and the optimization steps are performed on coarse grids only (Beux and Dervieux [9] proposed a different “one-shot” method which is done on a single computational grid but the number of control parameters is progressively increased performing a multilevel optimization). Ta’asan et al. [10] and Kuruvila et al. [11] applied these ideas to airfoil shape optimization using the full-potential equations. Arian [12] and Arian and Ta’asan [13],[14] extended the one-shot multigrid method to infinite dimensional design space in which the optimization steps are performed on all scales including the coarsest and the finest grids. However, the multigrid one-shot method requires major change in a given code before it can be applied and in many applications this is undesirable; one would like to use a method which is simple to include in a given code. This led to the development of new methods which are much simpler to apply and still are very effective. Ta’asan [15] proposed a single-grid method to accelerate the numerical convergence of boundary value optimization problems, known as the “pseudo-time” method. In the “pseudo-time” method the boundary conditions of the state and costate equations are maintained feasible, together with the design equation (defined on the boundary only), throughout the computation (see also [16]). Iollo et al. [17] applied the pseudo-time method to airfoil shape optimization governed by the Euler equations in two dimensions; the optimum is reported to be recovered in the computational effort of roughly 4 full solutions of the analysis problem for 4 and for 8 design variables.

One way to lower the computational cost is by approximating the cost functional Hessian and using the inverse of the Hessian to act on the gradient. The most popular method to approximate the Hessian, and its inverse, very effectively is the Quasi-Newton method (see for example Gill et al. [18]). However, Quasi-Newton methods are of low-rank and are difficult to apply efficiently for large ill-conditioned problems. The alternative to the Quasi-Newton method is a direct construction of the Hessian matrix. Haftka [19] introduced a method to compute the Hessian by solving  $N$  sensitivity equations, where  $N$  is the number of design variables, together with the adjoint equation. Jou et al. [20] use an approximation of such a method to construct the Hessian matrix for shape optimization of a three dimensional wing governed by the full-potential equation coupled with a boundary layer equation. Arian and Ta’asan [1] introduced a local mode analysis of the Hessian,  $H$ , for inviscid aerodynamic optimization problems and suggest a preconditioning method by transforming the Newton equation in Fourier space,  $\hat{H}\hat{s} = -\hat{g}$ , to the real space resulting in a preconditioning ODE for  $s$  ( $g$  here denotes the derivative of the cost functional with respect to the design variable). The analysis indicates that inviscid aerodynamic optimization problems are highly ill-conditioned and their condition number increases as the design space is refined; it is therefore necessary to approximate the Hessian, or rather the effect of its inverse on the gradient ( $s = -H^{-1}g$ ) in order to obtain effective convergence for such problems.

In this paper, the preconditioning method, suggested by Arian and Ta’asan [1] is re-derived for the compressible Euler equations in their conservative form in two space dimensions and is tested numerically for an optimal shape design problem in two dimensions. The design space is discretized in the maximum dimension, i.e., the location of each point on the intersection of the mesh with the shape is taken to be

an independent design variable. Thus, we take the continuous approach into its extreme, both in the approximations of the derivatives (gradient and Hessian) and in the discretization of the design space (in the continuous approach the derivation of the optimality conditions is done in the continuous, infinite-dimensional, level followed by discretization). The optimization algorithm is composed of solving numerically the Euler and their adjoint equations followed by a solution of the preconditioning ordinary differential equation (ODE) on the profile, and updating the profile's shape with that solution, resulting in a very efficient algorithm. We give numerical examples for 86 design variables in two different flow speeds and achieve an order of magnitude reduction in the cost functional at a computational effort of a full solution of the analysis PDE.

**2. Problem Definition.** Let  $\vec{U}$  denote the vector of state variables:

$$(2.1) \quad \vec{U} = (\rho, \rho u, \rho v, \rho w, \rho E)^T.$$

In terms of the Jacobian matrices,  $\vec{A} = (A, B)$ , defined in appendix A, the Euler equations are given by (conservative form)

$$(2.2) \quad \begin{aligned} \nabla(\vec{A}\vec{U}) &= 0 \quad \text{in } \Omega \\ \vec{u} \cdot \vec{n} &= 0 \quad \text{on } \Gamma, \end{aligned}$$

where  $\vec{u} = (u, v)$ ,  $\Gamma$  denotes the solid wall (airfoil), and with additional appropriate boundary conditions at the far-field.

The minimization problem in the continuum level is defined by

$$(2.3) \quad \min_{y(x)} F(p) = \oint_{\Gamma} (p(\xi) - p^*(\xi))^2 d\xi$$

subject to  $p$  satisfying the Euler equations in the domain  $\Omega$ . Note that the design variable is composed of two functions:  $y^U(x)$  and  $y^L(x)$  denoting the upper and lower sections of the airfoil. Fig.(1) depicts the geometry of the airfoil.

The pressure,  $p$ , depends on the fourth state variable,  $\rho E$ , by the relation (the parameter  $\gamma$  is set to 1.4)

$$(2.4) \quad p = (\gamma - 1)\rho E - \frac{\gamma - 1}{2}\rho q^2,$$

where  $q$  is the flow speed,  $q = \sqrt{u^2 + v^2}$ .

**3. The Adjoint Equations.** We derive the first order necessary conditions for a minimum in the continuum level. The derivation is done by taking a variation to the design variable  $y(\xi)$ ,

$$(3.1) \quad y^{n+1}(\xi) = y^n(\xi) + \varepsilon \tilde{\alpha}(\xi)$$

and computing its effect on the cost functional with the adjoint method:

$$(3.2) \quad F^{n+1} = F^n + \varepsilon \tilde{\alpha}(\xi) g(\xi).$$

The derivative  $g(\xi)$  is derived in appendix B:

$$(3.3) \quad g(\xi) = \frac{(\vec{y} \cdot \vec{n})}{R} (p - p^*)^2 + 2 \frac{\partial p}{\partial y} (p - p^*) - \left( \frac{\vec{\Lambda} \cdot \vec{U} + \lambda_4 p}{\rho} \right) \left( \frac{\partial \vec{u}}{\partial y} \cdot \vec{n} \right) - \frac{\partial}{\partial \xi} \left[ \left( \frac{\vec{\Lambda} \cdot \vec{U} + \lambda_4 p}{\rho} \right) (\vec{u} \cdot \vec{\xi})(\vec{y} \cdot \vec{n}) \right] = 0 \quad \text{on } \Gamma,$$

where  $R$  denotes the local radius of curvature,  $\vec{n}$  is a unit vector normal to the surface,  $\vec{y}$  is a unit vector in the vertical direction, and  $\vec{\Lambda}$  is the vector of costate variables,

$$(3.4) \quad \begin{aligned} \vec{\Lambda} &= (\lambda_1, \lambda_2, \lambda_3, \lambda_4)^T \\ \vec{\lambda} &= (\lambda_2, \lambda_3), \end{aligned}$$

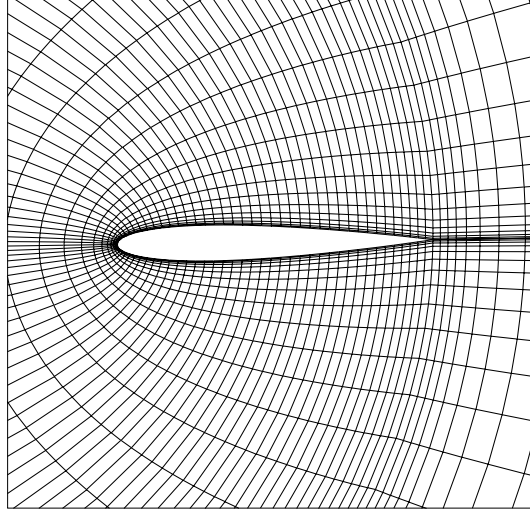


FIG. 1. The local coordinate system around the airfoil.

which satisfies the following “adjoint” equations (with additional far-field boundary conditions):

$$(3.5) \quad \begin{aligned} A^T \frac{\partial \vec{\lambda}}{\partial x} + B^T \frac{\partial \vec{\lambda}}{\partial y} &= 0 \quad \text{in } \Omega \\ \vec{\lambda} \cdot \vec{n} + 2(p - p^*) &= 0 \quad \text{on } \Gamma. \end{aligned}$$

**4. Local Mode Analysis of the Hessian.** We derive the symbol of the Hessian for the optimization problem, defined by Eqs.(2.2)-(2.3), along the same lines as done by Arian and Ta’asan [1]. The derivation is divided into subsections with the following content.

In §4.1 we find the relation between the state variables and the error in the shape. To achieve this we use the following representation of the error in the shape,  $\bar{\alpha}(\xi)$ ,

$$(4.1) \quad \bar{\alpha}(\xi) = \hat{\alpha}(\omega_1) e^{i\omega_1 \xi},$$

and the corresponding errors in the state variables,  $\{\bar{U}_j\}_{j=1}^4$ ,

$$(4.2) \quad \bar{U}_j = \sum_{k=1}^4 \beta_k(\omega_1) v_{kj} e^{i\omega_1 \xi} e^{i\omega_2 \eta}$$

where  $v_{kj}$  denotes the  $j$ ’th component of the null space eigenvector  $\vec{V}_k$ , and  $\beta_k$  are parameters.

In section §4.2 we find the relation between the symbol of the pressure variable and the symbol of the error in the shape variable.

In §4.3 we use the expression derived in §4.2 to compute the symbol of the Hessian.

**4.1. Null Space of the State Equation Jacobian.** We define the matrix  $L$  by

$$(4.3) \quad L\vec{U} = A \frac{\partial \vec{U}}{\partial x} + B \frac{\partial \vec{U}}{\partial y}.$$

Euler equations for fluids satisfy the relation  $p = \rho f(e)$ , where  $e = E - \frac{1}{2}q^2$ , and as a result  $\tilde{A}\vec{U} = 0$  (see for example Hirsch [21]). Therefore the error equations of (4.3) has the following form:

$$(4.4) \quad L\vec{U} = A \frac{\partial \vec{U}}{\partial x} + B \frac{\partial \vec{U}}{\partial y},$$

and the symbol of  $L$  is given by

$$(4.5) \quad \hat{L}(\omega_1, \omega_2) = i\omega_1 A + i\omega_2 B.$$

The solution of the linearized Euler error equations is spanned by four eigenvectors that correspond to four roots of the equation

$$(4.6) \quad \text{Det}(\hat{L}) = \frac{1}{2}\omega_1^2 u^2 \left[ 2\omega_1^2 u^2 + (\gamma - 1)\gamma(u^2 - 2E)(\omega_1^2 + \omega_2^2) \right] = 0.$$

The roots are given by  $\omega_1 = \{0, \pm \frac{1}{M^2 - 1}\}$ , where we use the relation  $M = q/c$  with

$$(4.7) \quad c^2 = \gamma(\gamma - 1) \left[ E - \frac{1}{2}q^2 \right].$$

For each of the four roots there corresponds an eigenvector,  $\vec{V}_k$ , for which  $\hat{L}\vec{V}_k = 0$ .

**4.1.1. The root  $\omega_1 = 0$ .** The two eigenvectors that correspond to the root  $\omega_1 = 0$  are given by

$$(4.8) \quad \begin{aligned} \vec{V}_1(\omega_1 = 0) &\equiv (v_{11}, v_{12}, v_{13}, v_{14})^T = \left(-\frac{2}{q^2}, 0, 0, 1\right)^T \\ \vec{V}_2(\omega_1 = 0) &\equiv (v_{21}, v_{22}, v_{23}, v_{24})^T = \left(\frac{2}{q}, 1, 0, 0\right)^T. \end{aligned}$$

We claim that these vectors do not play a role in the determination of the Hessian's symbol since they do not affect the pressure distribution on the boundary. In order to show this quantitatively we examine the relation (2.4) between the pressure,  $p$ , and the state variables. The error in  $p$  is given by

$$(4.9) \quad \bar{p} = (\gamma - 1) \left( \frac{q^2}{2} \bar{\rho} - u_1 \rho \bar{u}_1 - u_2 \rho \bar{u}_2 + \rho \bar{E} \right).$$

For  $\omega_1 = 0$  the contribution to the error in  $p$  by the first two eigenvectors is given by

$$(4.10) \quad (\gamma - 1) \left[ \left( \frac{q^2}{2} v_{11} - u_1 v_{12} - u_2 v_{13} + v_{14} \right) + \left( \frac{q^2}{2} v_{21} - u_1 v_{22} - u_2 v_{23} + v_{24} \right) \right] \hat{\alpha}(\omega_1, \omega_2) e^{i\omega_2 \eta}.$$

However, it is easy to check that

$$(4.11) \quad \begin{aligned} \frac{q^2}{2} v_{11} - u_1 v_{12} - u_2 v_{13} + v_{14} &= 0 \\ \frac{q^2}{2} v_{21} - u_1 v_{22} - u_2 v_{23} + v_{24} &= 0. \end{aligned}$$

**4.1.2. The root  $\omega_1 = -\frac{1}{M^2 - 1}$ .** The third eigenvector, which corresponds to the root  $\omega_2 = -\omega_1 \sqrt{M^2 - 1}$ , is not physical since it corresponds to a solution which is diverging in the subsonic regime and propagating into the  $-x$  direction in the supersonic regime:  $e^{i\omega_2 \eta} = e^{-i\omega_1 \sqrt{M^2 - 1} \eta}$  (for subsonic flow,  $M < 1$ , the above exponent diverges as  $\eta$  goes to the far field (for more details see Arian and Ta'asan [1])).

**4.1.3. The root  $\omega_1 = \frac{1}{M^2 - 1}$ .** We conclude that the only eigenvector that is physical and relevant to the Hessian's symbol is the fourth eigenvector that corresponds to the root  $\omega_2 = \omega_1 \sqrt{M^2 - 1}$ :

$$(4.12) \quad \vec{V}_4 \equiv (v_{41}, v_{42}, v_{43}, v_{44})^T = \frac{1}{den_V} \left( \frac{2(\gamma - 1)M^2}{q}, 2(\gamma - 1)(M^2 - 1), -2(\gamma - 1)\sqrt{M^2 - 1}, den_V \right)^T$$

where  $den_V = [2 + (M^2 - 2)(\gamma - 1)]q$ .

**4.2. The Symbol of the Pressure.** By the boundary condition on the solid wall

$$(4.13) \quad \vec{u} \cdot \vec{n} + \bar{\alpha} \left( \frac{\partial \vec{u}}{\partial y} \cdot \vec{n} \right) - \frac{\partial \bar{\alpha}}{\partial \xi} (\vec{u} \cdot \vec{\xi}) (\vec{n} \cdot \vec{y}) = 0.$$

Therefore the following holds for the coefficient  $\beta_4$  (see Eq. (4.2)):

$$(4.14) \quad \beta_4 = \frac{(i\omega_1 q(\vec{y} \cdot \vec{n}) - \frac{d\vec{u}}{dy} \cdot \vec{n}) \hat{\alpha}}{v_{43}}.$$

By Eq. (4.9)

$$(4.15) \quad \hat{p} = (\gamma - 1) \left( \frac{q^2}{2} v_{41} - q v_{42} + v_{44} \right) \beta_4 = q \frac{\frac{d\vec{u}}{dy} \cdot \vec{n} - i\omega_1 q(\vec{y} \cdot \vec{n})}{\sqrt{M^2 - 1}} \hat{\alpha}.$$

**4.3. Derivation of the Hessian's Symbol.** The cost functional (2.3) can be presented as an inner product in  $L^2(\Gamma)$ :

$$(4.16) \quad F = \langle p - p^*, p - p^* \rangle_{L^2(\Gamma)}.$$

In order to estimate the Hessian we need to expand the cost functional,  $F$ , as a Taylor series in a perturbation of the shape,  $\tilde{\alpha}$ . A perturbation of the shape,  $\tilde{\alpha}$ , results in a perturbed cost functional of the form

$$(4.17) \quad F(\alpha + \tilde{\alpha}) = \left\langle ((p - p^*) + \tilde{p} + \tilde{\alpha} \frac{\partial p}{\partial y}), ((p - p^*) + \tilde{p} + \tilde{\alpha} \frac{\partial p}{\partial y}) (1 + \frac{\tilde{\alpha}}{R}) \right\rangle_{L^2(\Gamma)} + h.o.t.$$

where  $R$  is local radius of curvature.

Let us assume that the operator  $\mathcal{Z}$  maps the change in the shape,  $\tilde{\alpha}$ , with the corresponding change in the pressure,  $\tilde{p}$ :

$$(4.18) \quad \tilde{p} = \mathcal{Z} \tilde{\alpha}.$$

In terms of  $\mathcal{Z}$  the Taylor expansion of (4.16) is given by:

$$(4.19) \quad \begin{aligned} F(\alpha + \tilde{\alpha}) &= F(\alpha) + \left\langle (p - p^*)(\mathcal{Z} + \mathcal{Z}^* + 2\frac{\partial p}{\partial y}), \tilde{\alpha} \right\rangle_{L^2(\Gamma)} \\ &+ \left\langle \left[ (\mathcal{Z}^* + \frac{\partial p}{\partial y})(\mathcal{Z} + \frac{\partial p}{\partial y}) + \frac{1}{R}(p - p^*)(2\frac{\partial p}{\partial y} + \mathcal{Z}^* + \mathcal{Z}) \right] \tilde{\alpha}, \tilde{\alpha} \right\rangle_{L^2(\Gamma)} + h.o.t. \end{aligned}$$

Therefore, in terms of  $\hat{\mathcal{Z}}$  the symbol of the Hessian is approximated by

$$(4.20) \quad \hat{H} = 2 \left( \hat{\mathcal{Z}}^* \hat{\mathcal{Z}} + \frac{\partial p}{\partial y} (\hat{\mathcal{Z}}^* + \hat{\mathcal{Z}}) + (\frac{\partial p}{\partial y})^2 + \frac{1}{R}(p - p^*)(2\frac{\partial p}{\partial y} + \hat{\mathcal{Z}}^* + \hat{\mathcal{Z}}) \right),$$

where by Eq.(4.15) the symbol of  $\mathcal{Z}$  is equal to

$$(4.21) \quad \hat{\mathcal{Z}} = q \frac{\frac{d\vec{u}}{dy} \cdot \vec{n} - i\omega_1 q(\vec{y} \cdot \vec{n})}{\sqrt{M^2 - 1}}.$$

We conclude that the symbol of the Hessian is approximated by

$$(4.22) \quad \hat{H} = a\omega^2 + b,$$

where  $a$  and  $b$  are given by

$$(4.23) \quad \begin{aligned} a &= 2 \left( \frac{q^4 (\vec{y} \cdot \vec{n})^2}{|M^2 - 1|} \right) \\ b &= 2 \left( (Re(\hat{\mathcal{Z}}) + \frac{\partial p}{\partial y})^2 + \frac{2}{R}(p - p^*)(\frac{\partial p}{\partial y} + Re(\hat{\mathcal{Z}})) \right). \end{aligned}$$

Transforming the Newton equation,  $\hat{H}\hat{s} = -\hat{g}$ , back into the real space we obtain the following preconditioning ODE, which approximates the Newton step, on the designed profile:

$$(4.24) \quad -a \frac{\partial^2 s}{\partial \xi^2} + bs = -g \quad \text{on } \Gamma.$$

**5. The Optimization Algorithm.** Starting with an approximation  $(\vec{U}^n(\Gamma^n), \vec{\Lambda}^n(\Gamma^n))$  (from here on the superscript  $n$  will be omitted)

1. Solve partially for  $\vec{U}^{n+1}$  the Euler equations (2.2) (reduce residuals by two orders of magnitude).
2. Solve partially for  $\vec{\Lambda}^{n+1}$  the adjoint equations (3.5), (reduce residuals by two orders of magnitude).
3. Compute the gradient,  $g$ , with Eq.(3.3).
4. Solve the Preconditioning equation (4.24) for  $s$ .
5. Update the shape with  $\Gamma^{new} = \Gamma^{old} + \delta s$  and re-mesh, where  $\delta$  is a parameter.
6. If the  $L^2$  norm of the gradient,  $\|g\|_{L^2(\Gamma)}$ , is smaller than some prescribed value then stop, else goto 1.

**6. Numerical Test.** A detailed description of the numerical scheme that we used for the solution of the Euler equations is given in Vatsa and Wedan [22] and in Vatsa et al. [23]. Here we give only a brief summary (the adjoint system (3.5) was solved in a similar manner to the solution of Euler system). We used a semi-discrete finite-volume algorithm based on Jameson’s Runge-Kutta time-stepping scheme (Jameson et al. [24] and Jameson and Baker [25]) to obtain the steady-state of the governing Euler and their adjoint equations. A controlled amount of artificial dissipation was added to the central-difference scheme to suppress odd-even point decoupling and oscillations in the vicinity of shock waves and stagnation points. The discrete design space was composed of the coordinates of the grid intersection with the profile,  $\{y_{i,1}\}_{i=1}^N$ , with  $N = 89$  (Fig.(1) depicts the grid around the airfoil). The leading edge and two trailing edge grid points were fixed during the computation and therefore the actual number of design variables was 86. We minimize the pressure matching cost functional, (2.3), where the discrete target pressure distribution,  $\{p^*\}_{i=1}^N$ , was computed on the NACA 0012 airfoil, and the initial geometry was taken to be the Onera M6 airfoil.

Our assumption is that in the advanced phase of aircraft design the wing’s planform is fixed and therefore we fixed the leading and trailing edges of the airfoil. The airfoil was divided into two sections: the upper part  $y^U(x)$  and the lower part  $y^L(x)$ . On each section we used 43 design variables, having a total number of 86 design variables, and the optimization algorithm presented in Sec.6 was applied.

Given the discretized gradient along the airfoil,  $\{g_i^h\}_{i=1}^N$ , we solve the discrete version of Eq.(4.24) for  $\{s_i^h\}_{i=1}^N$  on the airfoil, with the boundary conditions  $s^h = 0$  on the trailing and leading edges (our numerical experiments show that setting  $b = 0$  in Eq.(4.24) results in a more robust preconditioner, although less accurate in some cases). Thus Eq.(4.24) is solved twice, i.e., once on the upper section of the airfoil and once on the lower section. At transonic flow, where there are sonic points along the airfoil we introduce a cutoff on the denominator of Eq.(4.24):  $|1 - M^2| \geq \epsilon$  (we used  $\epsilon = 0.4$ ). The solution of Eq.(4.24),  $\{s_i^h\}_{i=1}^N$ , is the new “preconditioned” search direction.

Two flow speeds were tested: subsonic with  $M = 0.5$  and transonic with  $M = 0.75$ .

**6.1. Subsonic Flow.** Table 6.1 depicts the cost functional history in logarithmic scale for Mach number 0.5. At each optimization step we used a fixed number of iterations (50) for the flow and adjoint partial solutions. We observe an average reduction of about 0.23 in the cost functional at every iteration, while the computational effort required by each iteration is of about a third of one cost functional evaluation (50



iterations for the state equations and another 50 iterations for the costate equations, while it takes about 300 iterations to solve the Euler equations to machine zero accuracy (one cost functional evaluation)).

Fig.2 depicts the error (err), preconditioned solution (s), and gradient (g), for the first four iterations (all normalized). The preconditioning equation (4.24) transformed the gradient (g) into a new function (s) which approximates the actual error in the shape (err). The actual error in the shape is computed by  $\{err_i^h = y_i^{target} - y_i^h\}_{i=1}^N$  where  $y_i^{target}$  denotes the  $y$  coordinates at the mesh point  $i$  along the boundary of the NACA 0012 airfoil, and  $y_i^h$  denotes the  $y$  coordinates of the current iterate airfoil (starting with the Onera M6 airfoil).

| Iteration       | 1                    | 2                    | 3                    | 4                    | 5                    |
|-----------------|----------------------|----------------------|----------------------|----------------------|----------------------|
| Cost Functional | $4.32 \cdot 10^{-4}$ | $1.17 \cdot 10^{-4}$ | $2.16 \cdot 10^{-5}$ | $6.04 \cdot 10^{-6}$ | $1.28 \cdot 10^{-6}$ |

TABLE 6.1

*Cost functional history for Mach number 0.5. The initial airfoil is Onera M6 and the target pressure distribution is taken from a NACA 0012 airfoil.*

**6.2. Transonic Flow.** The results for the transonic case are depicted in table 6.2 and Fig.3.

Table 6.2 depicts the cost functional history in logarithmic scale for Mach number 0.75. (The initial airfoil is Onera M6 and the target pressure distribution is taken from a 0012 airfoil.) At each optimization step we used a fixed number of iterations (100) for the flow and adjoint partial solutions. In this case the convergence rate is slower than in the subsonic flow; we observe an average reduction of about 0.33 in the cost functional at every iteration, while the computational effort required by each iteration is of about a two thirds of one cost functional evaluation.

Fig.3 depicts the error (err), preconditioned (s), and gradient (g), for the first four iterations (all normalized).

| Iteration       | 1                    | 2                    | 3                    | 4                    | 5                    |
|-----------------|----------------------|----------------------|----------------------|----------------------|----------------------|
| Cost Functional | $6.52 \cdot 10^{-3}$ | $2.04 \cdot 10^{-3}$ | $5.04 \cdot 10^{-4}$ | $2.20 \cdot 10^{-4}$ | $7.64 \cdot 10^{-5}$ |

TABLE 6.2

*Cost functional history for Mach number 0.75. The initial airfoil is Onera M6 and the target pressure distribution is taken from a NACA 0012 airfoil.*

**7. Discussion and Concluding Remarks.** An important issue in the field of numerical methods for optimization problems governed by PDEs is the discrete versus the continuum approaches. In the discrete approach, the derivation of the first and second order sensitivities is done in the discrete level (for example with automatic differentiation), while in the continuum approach the derivation is done in the continuous, infinite-dimensional, level followed by discretization. In this paper we take the continuous approach to its extreme; both in the approximations of the derivatives (gradient and Hessian) and in the discretization of the design space. The main advantage of the continuous approach is that we can use the structure of the problem by analyzing it in the PDE level, resulting in a very efficient computational method, as demonstrated in this paper. However, we find that the continuous approach has its disadvantages. We had some difficulties with the point-wise control of the shape, and with the sensitivity gradients accuracy, around singular points. In our opinion, there is a tradeoff between robustness and efficiency of the numerical scheme, where the continuous approach leads to schemes which are less robust but much more computationally efficient than the schemes resulting from the discrete approach.

## REFERENCES

- [1] E. ARIAN AND S. TA'ASAN, *Analysis of the Hessian for Aerodynamics Optimization: Inviscid Flow*, ICASE report No. 96-28, (1996) (<http://www.icase.edu/~arian/pubs.html>).
- [2] M. J. LIDTHILL, *A new method of two dimensional aerodynamic design*, R & M 1111, Aeronautical Research Council (1945).
- [3] O. PIRONNEAU, *On Optimum Design in Fluid Mechanics*, Journal of Fluids Mechanics, Vol. 64 (1974).
- [4] A. JAMESON, *Aerodynamic Design Via Control Theory*, ICASE report No. 88-64, Journal of Scientific Computing. **3**, 233 (1988).
- [5] B. MANDEL, J. PERIAUX, B. STOUFFLET, *Optimum Design Methods in Aerodynamics*, AGARD-FDP-VKI Special Course (1994).
- [6] B. I. SOEMARWOTO, *Multi-Point Aerodynamic Design by Optimization*, Ph.D. thesis, Delft University of Technology, the Netherlands (1996).
- [7] A. JAMESON, N. A. PIERCE, AND L. MARTINELLI, *Optimum Aerodynamic Design using the Navier-Stokes Equations*, AIAA-97-0101, 35th Aerospace Sciences Meeting & Exhibit (1997).
- [8] S. TA'ASAN, *"One Shot" Methods for Optimal Control of Distributed Parameter Systems I: Finite Dimensional Control*, ICASE report NO. 91-2 (1991).
- [9] F. BEUX AND A. DERVIEUX, *A Hierarchical Approach for Shape Optimization*. Technical Report 1868, INRIA (1993).
- [10] S. TA'ASAN, G. KURUVILA, M. D. SALAS, *Aerodynamic Design and Optimization in One Shot*, 30th Aerospace Sciences Meeting & Exhibit, AIAA 92-0025 (1992).
- [11] G. KURUVILA, S. TA'ASAN, M. D. SALAS, *Airfoil Optimization by the One-Shot Method*, *Optimum Design Methods in Aerodynamics*, AGARD-FDP-VKI Special Course (1994).
- [12] E. ARIAN, *Multigrid Methods for Optimal Shape Design Governed by Elliptic Systems*, Ph.D. Thesis, The Weizmann Institute of Science, Israel (1994).
- [13] E. ARIAN AND S. TA'ASAN, *Multigrid One Shot Methods for Optimal Control Problems: Infinite Dimensional Control*, ICASE report No. 94-52 (1994).
- [14] E. ARIAN AND S. TA'ASAN, *Smoothers for Optimization Problems*, Proceedings of the Seventh Copper Mountain Conference on Multigrid Methods (1995).
- [15] S. TA'ASAN, *Pseudo-Time Methods for Constrained Optimization Problems Governed by PDE*, ICASE Report 95-32 (1995).
- [16] S. TA'ASAN, *Trends in Aerodynamics Design and Optimization: a mathematical view point*, Proceedings of the 12th AIAA Computational Fluid Dynamics Conference, AIAA-95-1731, pp.961-970 (1995).
- [17] A. IOLLO, G. KURUVILA, AND S. TA'ASAN, *Pseudo-Time Method for Optimal Shape Design Using the Euler Equations*, ICASE report No. 95-59 (1995).
- [18] P. E. GILL, W. MURRAY, AND M. H. WRIGHT, *Practical Optimization*, Academic Press INC (1981).
- [19] R. T. HAFTKA, *Second Order Design Derivatives in Structural Analysis*, AIAA Journal **20**, 1765-1766 (1982).
- [20] W. H. JOU, W. P. HUFFMAN, D. P. YOUNG, R. G. MELVIN, M. B. BIETERMAN, C. L. HILMES, AND F. T. JOHNSON, *Practical Considerations in Aerodynamic Design and Optimization*, Proceedings of the 12th AIAA Computational Fluid Dynamics Conference, AIAA-95-1730, pp.950-960 (1995).

- [21] C. HIRSCH, *Numerical Computation of Internal and External Flows*, Vol.2, John Wiley & Sons, pp.139 (1988).
- [22] V. N. VATSA AND B. W. WEDAN, *Development of a Multigrid Code for 3-D Navier-Stokes Equations and its Application to a Grid-Refinement Study*, Computers & Fluids Vol. 18, No. 4, pp. 391-403 (1990).
- [23] V. N. VATSA, M. D. SANETRIK, AND E. B. PARLETTE, *Development of a Flexible and Efficient Multigrid-Based Multiblock Flow Solver*, AIAA Paper 93-0677 (1993).
- [24] A. JAMESON, W. SCHMIDT, AND E. TURKEL, *Numerical Solutions of the Euler Equations by Finite Volume Methods using Runge-Kutta time Stepping Schemes*, AIAA Paper 81-1259 (1981).
- [25] A. JAMESON AND T. J. BAKER, *Solution of the Euler Equations for Complex Configurations*, AIAA Paper 83-1929 (1983).

#### Appendix A. Definition of the Euler Jacobian Matrices.

$$(A.1) \quad A = \begin{pmatrix} 0 & 1 & 0 & 0 \\ -u^2 + \frac{\gamma-1}{2}q^2 & (3-\gamma)u & -(\gamma-1)v & \gamma-1 \\ -uv & v & u & 0 \\ -u(\gamma E - (\gamma-1)q^2) & \gamma E - \frac{\gamma-1}{2}(q^2 + 2u^2) & -(\gamma-1)uv & \gamma u \end{pmatrix}$$

$$(A.2) \quad B = \begin{pmatrix} 0 & 0 & 1 & 0 \\ -uv & v & u & 0 \\ -v^2 + \frac{\gamma-1}{2}q^2 & -(\gamma-1)u & (3-\gamma)v & \gamma-1 \\ -v(\gamma E - (\gamma-1)q^2) & -(\gamma-1)uv & \gamma E - \frac{\gamma-1}{2}(q^2 + 2v^2) & \gamma v \end{pmatrix}.$$

**Appendix B. Derivation of the Shape Derivatives.** We compute the derivative of the cost functional with respect to the shape of the profile,  $y(\xi)$ , with the adjoint method in the continuous level. A Lagrangian is defined as follows:

$$(B.1) \quad \mathcal{L}(\vec{U}, \vec{\Lambda}) = \oint_{\Gamma} (p - p^*)^2 d\xi + \int_{\Omega} \left[ \vec{\Lambda} \cdot \left( A \frac{\partial \vec{U}}{\partial x} + B \frac{\partial \vec{U}}{\partial y} \right) \right] dx dy.$$

Then an arbitrary perturbation of the profile's shape,  $\tilde{\alpha}$ , is introduced with a small amplitude  $\varepsilon$ :

$$(B.2) \quad y(x)^{new} = y(x)^{old} + \varepsilon \tilde{\alpha}(x)$$

The resulting first order variation of the Lagrangian, is given by (see for example [3], [12] for further details)

$$(B.3) \quad \begin{aligned} \tilde{\mathcal{L}} = & \oint_{\Gamma} \left[ 2(\tilde{p} + \tilde{\alpha} \frac{\partial p}{\partial y})(p - p^*) + \tilde{\alpha} \frac{(p - p^*)^2}{R} (\vec{y} \cdot \vec{n}) \right] d\xi + \int_{\Omega} \left[ \tilde{\vec{U}} \cdot \left( A^T \frac{\partial \vec{\Lambda}}{\partial x} + B^T \frac{\partial \vec{\Lambda}}{\partial y} \right) \right] dx dy \\ & + \int_{\partial\Omega} \vec{\Lambda}^T \left[ (n_1 A + n_2 B) \tilde{\vec{U}} \right] d\xi, \end{aligned}$$

where  $n_1$  and  $n_2$  are the components of the unit normal vector in the  $x$  and  $y$  directions, respectively, and where we have used the following relation (see [3])

$$(B.4) \quad d\xi^{new} = d\xi^{old} \left( 1 + \frac{\tilde{\alpha}}{R} (\vec{y} \cdot \vec{n}) \right).$$

Next we compute the term  $\vec{\Lambda}^T (n_1 A + n_2 B) \tilde{U}$  on the solid wall. The no-slip boundary condition  $\vec{u} \cdot \vec{n} = 0$  implies

$$(B.5) \quad \tilde{u} \cdot \vec{n} + \tilde{\alpha} \left( \frac{\partial \tilde{u}}{\partial y} \cdot \vec{n} \right) - \frac{\partial \tilde{\alpha}}{\partial \xi} (\vec{y} \cdot \vec{n}) (\vec{u} \cdot \vec{\xi}) = 0 \quad \text{on } \Gamma,$$

where we have used the following relation

$$(B.6) \quad \vec{n}^{new} = \vec{n}^{old} - \frac{\partial \tilde{\alpha}}{\partial \xi} (\vec{y} \cdot \vec{n}) \vec{\xi}.$$

Using relation (B.5) in  $\tilde{U}$  on the solid wall, and the variational form of Eq.(2.4), results in the following equation:

$$(B.7) \quad \vec{\Lambda}^T (n_1 A + n_2 B) \tilde{U} = (\vec{\lambda} \cdot \vec{n}) \tilde{p} - \frac{(\vec{\Lambda} \cdot \vec{U} + \lambda_4 p)}{\rho} \left( \left( \frac{\partial \tilde{u}}{\partial y} \cdot \vec{n} \right) \tilde{\alpha} - \frac{\partial \tilde{\alpha}}{\partial \xi} (\vec{y} \cdot \vec{n}) (\vec{u} \cdot \vec{\xi}) \right) \quad \text{on } \Gamma.$$

Equating the terms in  $\tilde{\mathcal{L}}$  that multiply  $\tilde{u}$  on  $\Gamma$  to zero results in the solid wall boundary condition for the adjoint problem:

$$(B.8) \quad \vec{\lambda} \cdot \vec{n} + 2(p - p^*) = 0 \quad \text{on } \Gamma.$$

Equating the terms in  $\tilde{\mathcal{L}}$  that multiply  $\tilde{\alpha}$  on  $\Gamma$  to zero results in the design equation:

$$(B.9) \quad g(\xi) = \frac{(\vec{y} \cdot \vec{n})}{R} (p - p^*)^2 + 2 \frac{\partial p}{\partial y} (p - p^*) - \left( \frac{\vec{\Lambda} \cdot \vec{U} + \lambda_4 p}{\rho} \right) \left( \frac{\partial \tilde{u}}{\partial y} \cdot \vec{n} \right) - \frac{\partial}{\partial \xi} \left[ \left( \frac{\vec{\Lambda} \cdot \vec{U} + \lambda_4 p}{\rho} \right) (\vec{u} \cdot \vec{\xi}) (\vec{y} \cdot \vec{n}) \right] = 0 \quad \text{on } \Gamma.$$

## Subsonic Flow

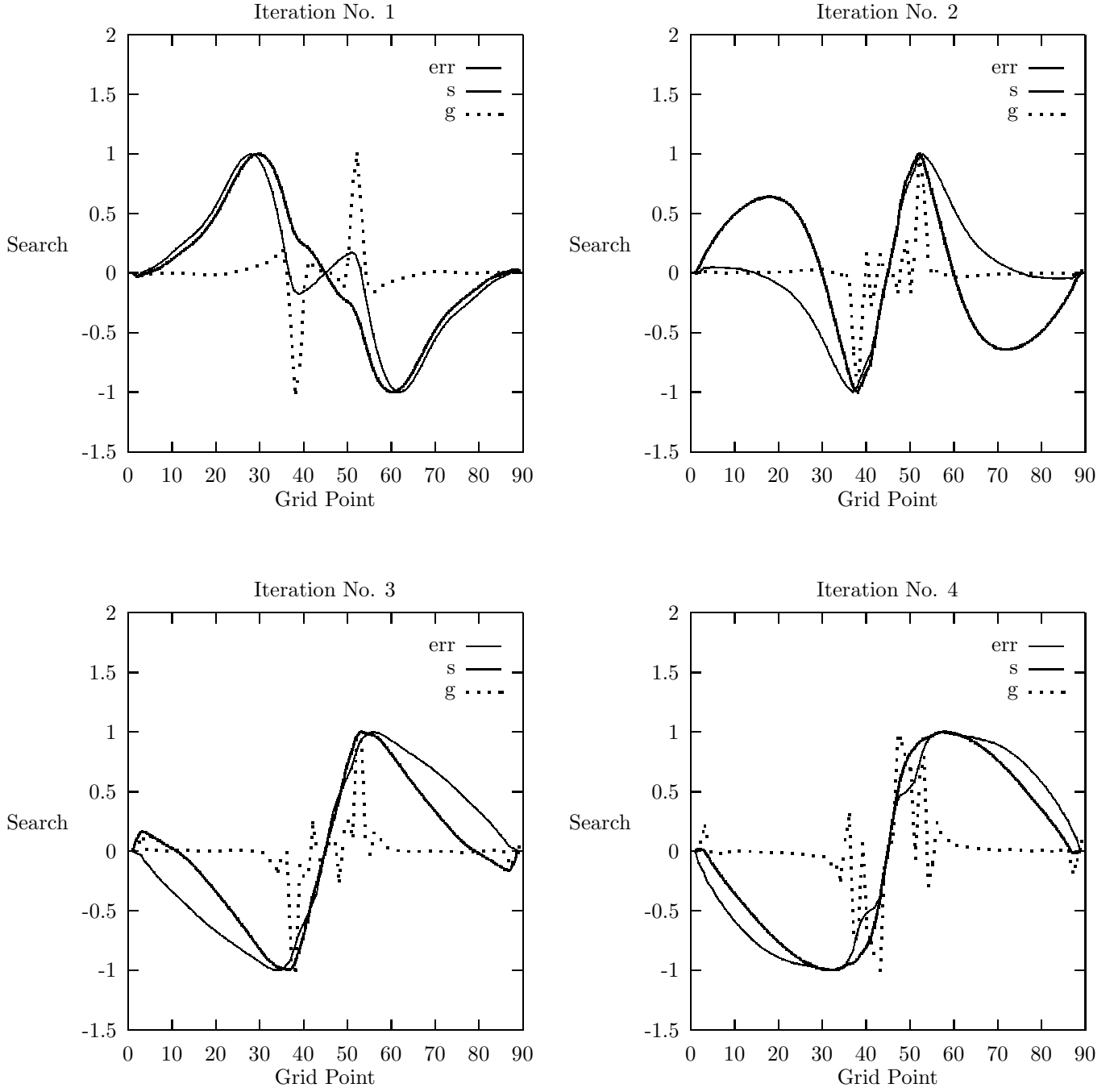


FIG. 2. The vertical axis represents the normalized actual error ( $err$ ) in the airfoil's shape, the normalized estimated error ( $s$ ), with the new preconditioning method, and the normalized sensitivity gradient ( $g$ ) of a cost functional with respect to the shape parameters along the airfoil. The horizontal axis represents the location of the computational mesh points on the airfoil.

# Transonic Flow

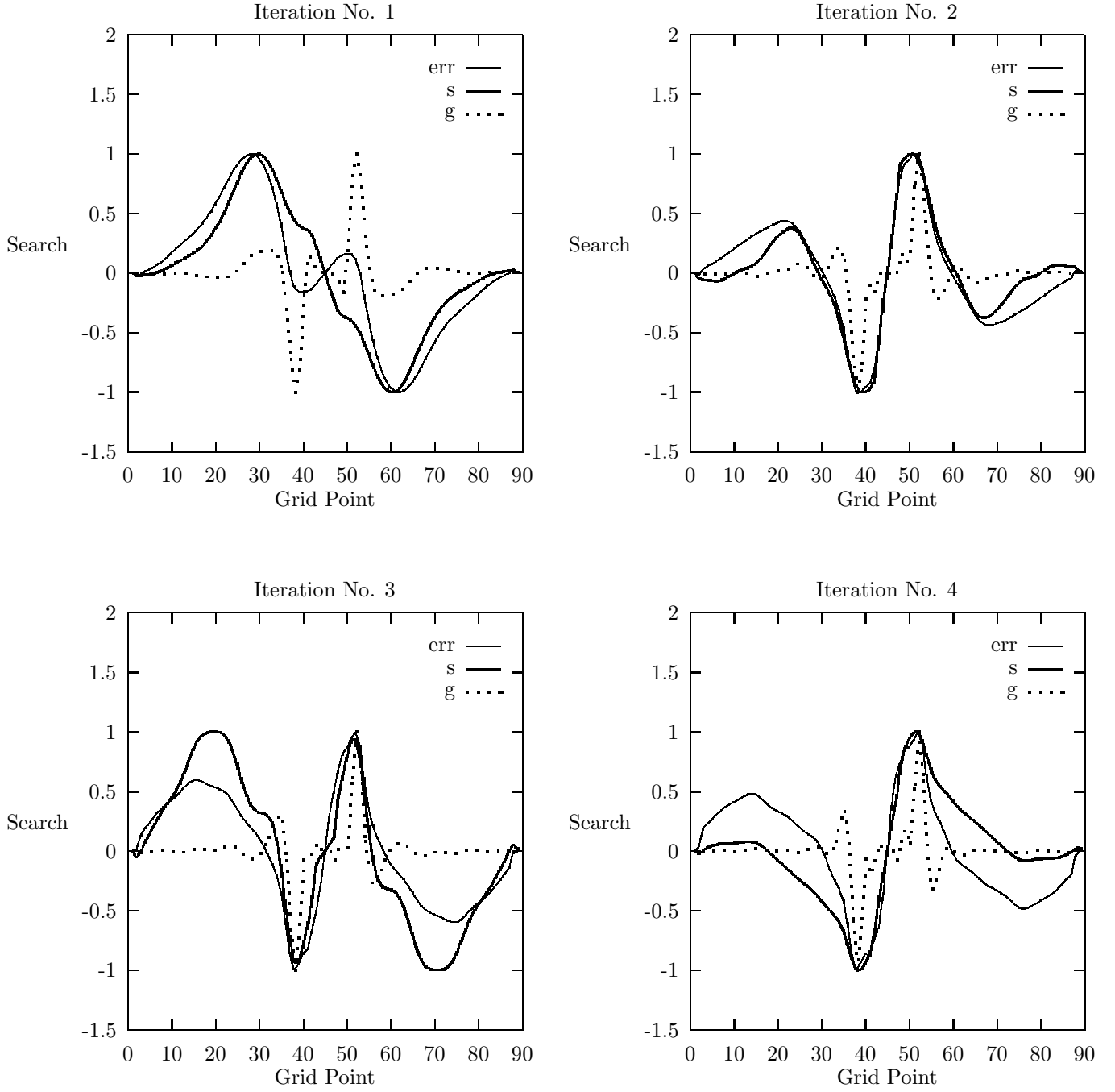


FIG. 3. The normalized error ( $err$ ), preconditioned solution ( $s$ ), and gradient ( $g$ ), for the first four iterations (starting with *Onera M6* with  $M = 0.75$ ).